V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

ROBUST IMPLEMENTATION OF THE SPALART–ALLMARAS TURBULENCE MODEL FOR UNSTRUCTURED GRID

Nikhil Vijay Shende* and Yair Mor–Yossef†

*Computational Aerodynamics Laboratory, Aerospace Engineering,
Indian Institute of Science, Bangalore 560012, India
e–mail: nikvijay@aero.iisc.ernet.in

†Israeli CFD Center
Caesarea Industrial Park, 38900 Israel
e–mail: yairm@iscfdc.co.il

Key words: Unstructured Mesh, Spalart–Allmaras turbulence model, Positive scheme, M–matrix, High lift configurations

Abstract. *In the present work, we have proposed a robust implementation of Spalart–Allmaras turbulence model for unstructured grid using a positive implicit procedure. The implicit procedure is based on designing the associated implicit matrix such that it is M–matrix. The implicit procedure employs a unified treatment for the implicit operator of convection, diffusion, anti–diffusion and source terms involved in Spalart–Allmaras model equation. This implicit procedure guarantees positivity of modified turbulent viscosity ($\tilde{\nu}$) without the use of any clipping. The efficacy of the implicit procedure is demonstrated with the help of two high lift configurations. From the results presented in this paper, it is evident that the present implicit procedure is capable of not only achieving high level of convergence for turbulent viscosity but also using large CFL number to accelerate convergence to steady state.*

1 INTRODUCTION

In past few years, Computational Fluid Dynamics (CFD) has reached to a stage of maturity where it can be used as an effective analysis tool in an industrial design cycle. The unstructured data based finite volume algorithms are not only proving to be accurate but also provide the designers with the reliable estimates of design data in a time short enough to impact the design cycle.

Many flow analysis problems of industrial relevance involve solving Reynolds Averaged Navier–Stokes (RANS) equations with an appropriate turbulence model. The use of turbulence models involving one or more equations, such as Spalart–Allmaras [1], k-ω and its variants [2, 3, 4] are common in Aerospace industry. Turbulence model equations are in the form of non–linear convection–diffusion equations with stiff source terms. The numerical stiffness associated with these equations not only restrict the choice of time step during solution evolution to steady state but also may cause solution divergence. This problem is further accentuated on fine grids employing very thin viscous padding around the body to accurately capture viscous sub–layer as well as solution adaptive grids required to accurately capture features like shocks, shear layers, vortices etc. Hence it becomes important to discretize the turbulence model equations, both in space and time, in such a way that the resultant system of equations preserves the positivity of the turbulent quantities during solution evolution.

Recently, Mor–Yossef and Levy [5] have proposed a new general implicit procedure that preserves the positivity of dependent variables corresponding to turbulence model equations. The implicit procedure is based on designing the implicit matrix to be M–matrix. The M–matrix has certain properties (to be discussed in section 2) desirable to construct the unconditionally positive implicit scheme. The implicit procedure employs a unified treatment for the implicit operator of convection, diffusion and source terms involved in model equations. This implicit procedure guarantees positivity of turbulence quantities without the use of any clipping. They have demonstrated the efficacy of this discretization procedure using variety of turbulence models [5, 6] both for steady and unsteady flow problems [7]. In the present work, we have extended the scope this procedure to Spalart–Allmaras turbulence model which is one of the most preferred turbulence models in Aerospace CFD community.

The organization of this paper is as follows. Section 2 presents the methodology behind the construction of unconditionally positive implicit procedure for Spalart–Allmaras turbulence model. Section 3 presents the specific form of the implicit matrix that leads to positive implicit formulation. Section 4 presents results and discussion. Finally, section 5 concludes the present work.

2 METHODOLOGY

The semi–discrete form of the Spalart–Allmaras turbulence model equation for a finite volume (cell) i can be written as:

$$\frac{d\tilde{\nu}_i}{dt} = R_i + \Omega_i \; S_i \tag{1}$$

where, for cell i, modified turbulence viscosity is denoted by $\tilde{\nu}_i$, sum of inviscid, viscous and anti–diffusion fluxes is denoted by R_i, source term is denoted by S_i, time is denoted by t and cell volume is denoted by Ω_i. Using first order backward Euler time stepping procedure to discretize Eq. 1 in time, following vector–matrix equation can be written for all cells in the computational domain:

$$\left[\frac{\Omega}{\Delta t}\mathcal{I} - \frac{\partial R}{\partial \tilde{\nu}} - \Omega \frac{\partial S}{\partial \tilde{\nu}}\right]^n \Delta \; \tilde{\nu}^{\,n} = R^n + \Omega S^n \tag{2}$$

where $\mathcal{I}$ is the identity matrix, n is the current time level, $\Delta\,(.) = (.)^{n+1} - (.)^n$. Let a matrix $\mathcal{M}$ be an approximation to $-\left(\frac{\partial R}{\partial \tilde{\nu}} + \Omega \frac{\partial S}{\partial \tilde{\nu}}\right)$ such that it fulfills the following two conditions:

1. $\mathcal{M}$ is an M–matrix,
2. $R^n + \Omega S^n + \mathcal{M}\tilde{\nu}^n$ is a non–negative vector.

It should be noted that the first condition guarantees only the convergence of the system of equations but not its positivity, i.e., the system of equations may converge to negative solution. With the addition of second condition one can guarantee non–negative solution of this system.

Substituting the matrix $-\left(\frac{\partial R}{\partial \tilde{\nu}} + \Omega \frac{\partial S}{\partial \tilde{\nu}}\right)$ in Eq. (2) by the matrix $\mathcal{M}$, we get:

$$\left[\frac{\Omega}{\Delta t}\mathcal{I} + \mathcal{M}\right]^n \Delta\tilde{\nu}^n = (R + \Omega S)^n \tag{3}$$

The following properties of the M–matrix are pivotal in the design of the proposed positive implicit scheme:

1. The inverse of an M–matrix exists,
2. The inverse of an M–matrix is non–negative diagonal positive (NPD) matrix with all diagonal entries positive and all off–diagonal entries non–negative,
3. For any positive scalar φ and for an M–matrix, say matrix $\mathcal{A}$, the matrix $\varphi\mathcal{I} + \mathcal{A}$ is an M–matrix.

It can be easily verified that the implicit scheme given by Eq. (3) is an unconditionally positive convergent scheme.

3 UNCONDITIONALLY POSITIVE FORMULATION

A specific form of the desired $\mathcal{M}$ for the Spalart–Allmaras model is developed herein. For this purpose, it is convenient to split the vector R_i given in Eq. 1 into inviscid, viscous and anti–diffusion parts as follows:

$$R_i \equiv R_{inv} + R_{vis} + R_{ad} \tag{4}$$

where R_{inv}, R_{vis} and R_{ad} denote inviscid, viscous and anti–diffusion residuals respectively. These residuals are given as follows:

$$\begin{aligned} R_{inv} &= -\sum_J F_{\perp J,inv}\ \Delta S_J &(5)\\ R_{vis} &= \sum_J F_{\perp J,vis}\ \Delta S_J &(6)\\ R_{ad} &= -\sum_J F_{\perp J,ad}\ \Delta S_J. &(7)\end{aligned}$$

In above equations, $F_{\perp J,inv}, F_{\perp J,vis}$ and $F_{\perp J,ad}$ denote respectively inviscid, viscous and anti–diffusion flux vectors normal to interface J and are given by

$$\begin{aligned} F_{\perp J,inv} &= u^+_{\perp i}\ \tilde{\nu}_i + u^-_{\perp j}\ \tilde{\nu}_j, &(8)\\ F_{\perp J,vis} &= \frac{1}{2\sigma}\left[(\nu_i+\nu_j) + (1+c_{b_2})(\tilde{\nu}_i+\tilde{\nu}_j)\right](\nabla\tilde{\nu}\cdot\hat{n})_J\,, &(9)\\ F_{\perp J,ad} &= \frac{c_{b_2}}{\sigma}\tilde{\nu}_i(\nabla\tilde{\nu}\cdot\hat{n})_J\,, &(10)\end{aligned}$$

with

$$\begin{aligned} u_{\perp i,j} &= u_{i,j}\ n_x + v_{i,j}\ n_y + w_{i,j}\ n_z, &(11)\\ u^\pm_\perp &= \frac{1}{2}\left[u_\perp \pm |u_\perp|\right] \text{and} &(12)\\ (\nabla\tilde{\nu}\cdot\hat{n})_J &= \frac{(\tilde{\nu}_j-\tilde{\nu}_i)}{\left|\vec{R}_{ij}\cdot\hat{n}_J\right|}. &(13)\end{aligned}$$

In above equations, $\vec{R}_{ij}$ denotes the distance between the cell centroids i and j shared by interface J and $\hat{n}_J$ denotes the unit normal to the interface J.

3.1 Presentation of inviscid residual

Substituting Eq. 8 in Eq. 5, the inviscid residual is written as follows:

$$\begin{aligned} R_{inv} &= -\sum_J \left(u^+_{\perp i}\ \tilde{\nu}_i + u^-_{\perp j}\ \tilde{\nu}_j\right)\Delta S_J \\ &= -\sum_J u^+_{\perp i}\Delta S_J\ \tilde{\nu}_i - \sum_J u^-_{\perp j}\Delta S_J\ \tilde{\nu}_j \\ &= -R_{i,inv}\ \tilde{\nu}_i - \sum_J R_{j,inv}\ \tilde{\nu}_j \end{aligned} \tag{14}$$

Consider following splitting for a scalar X

$$(X)_P = |X| + X \text{ and} \tag{15}$$

$$(X)_N = |X| - X. \tag{16}$$

Since $R_{i,inv}$ is always ≥ 0, we can write

$$(R_{i,inv})_P = 0 \text{ and} \tag{17}$$

$$(R_{i,inv})_N = R_{i,inv}. \tag{18}$$

Further, since $R_{j,inv}$ is always ≤ 0, we can write

$$(R_{j,inv})_P = -R_{j,inv} \text{ and} \tag{19}$$

$$(R_{j,inv})_N = 0. \tag{20}$$

Therefore,

$$R_{inv} = -\left(R_{i,inv}\right)_N\ \tilde{\nu}_i + \sum_J \left(R_{j,inv}\right)_P \tilde{\nu}_j. \tag{21}$$

3.2 Presentation of viscous residual

Substituting Eq. 13 and Eq. 9 in Eq. 6, the viscous residual is written as follows:

$$\begin{aligned} R_{vis} &= \sum_J \frac{1}{2\sigma}\left[(\nu_i+\nu_j) + (1+c_{b2})\left(\tilde{\nu}_i+\tilde{\nu}_j\right)\right]\frac{\left(\tilde{\nu}_j-\tilde{\nu}_i\right)}{\left|\vec{R}_{ij}\cdot\hat{n}_J\right|}\Delta S_J \\ &= \sum_J \Gamma_{vis}\left(\tilde{\nu}_j-\tilde{\nu}_i\right)\Delta S_J, \text{ with} \end{aligned} \tag{22}$$

$$\Gamma_{vis} = \frac{1}{2\sigma}\left[(\nu_i+\nu_j) + (1+c_{b2})\left(\tilde{\nu}_i+\tilde{\nu}_j\right)\right]\frac{1}{\left|\vec{R}_{ij}\cdot\hat{n}_J\right|} \tag{23}$$

Therefore,

$$\begin{aligned} R_{vis} &= -\sum_J \Gamma_{vis} \Delta S_J \ \tilde{\nu}_i + \sum_J \Gamma_{vis} \Delta S_J \ \tilde{\nu}_j \\ &= -R_{i,vis} \ \tilde{\nu}_i + \sum_J R_{j,vis} \ \tilde{\nu}_j \\ &= -\left(R_{i,vis}\right)_N \tilde{\nu}_i + \sum_J \left(R_{j,vis}\right)_P \tilde{\nu}_j \end{aligned} \tag{24}$$

3.3 Presentation of anti–diffusion residual

Substituting Eq. 13 and Eq. 10 in Eq. 7, the anti–diffusion residual is given by:

$$\begin{aligned} R_{ad} &= -\sum_J \frac{c_{b_2}}{\sigma} \ \tilde{\nu}_i \frac{(\tilde{\nu}_j - \tilde{\nu}_i)}{\left|\vec{R}_{ij} \cdot \hat{n}_J\right|} \Delta S_J \\ &= -\sum_J \Gamma_{ad} \left(\tilde{\nu}_j - \tilde{\nu}_i\right) \Delta S_J, \ \text{with} \end{aligned} \tag{25}$$

$$\Gamma_{ad} = \frac{c_{b_2}}{\sigma} \ \tilde{\nu}_i \frac{1}{\left|\vec{R}_{ij} \cdot \hat{n}_J\right|} \tag{26}$$

Therefore,

$$\begin{aligned} R_{ad} &= \sum_J \Gamma_{ad} \Delta S_J \ \tilde{\nu}_i - \sum_J \Gamma_{ad} \Delta S_J \ \tilde{\nu}_j \\ &= R_{i,ad} \ \tilde{\nu}_i - \sum_J R_{j,ad} \ \tilde{\nu}_j \\ &= \left(R_{i,ad}\right)_P \tilde{\nu}_i - \sum_J \left(R_{j,ad}\right)_N \tilde{\nu}_j \end{aligned} \tag{27}$$

3.4 Presentation of total residual

Based on the presentations of inviscid, viscous and anti–diffusion residuals, the total residual R_i may now reads as follows:

$$\begin{aligned} R_i = &- \left[\left(R_{i\,,inv}\right)_N + \left(R_{i\,,vis}\right)_N - \left(R_{i\,,ad}\right)_P\right] \tilde{\nu}_i \\ &+ \sum_J \left\{ \left[\left(R_{j\,,inv}\right)_P + \left(R_{j\,,vis}\right)_P - \left(R_{j\,,ad}\right)_N \right] \tilde{\nu}_j \right\} \end{aligned} \tag{28}$$

At this point its should be re-emphasized that $(R_{*,\,*})_P$, $(R_{*,\,*})_N$ are positive. Furthermore, one should note that:

$$\left(R_{i,vis}\right)_N = \sum_J \left(R_{j,vis}\right)_P \tag{29}$$

$$\left(R_{i,ad}\right)_P = \sum_J \left(R_{j,ad}\right)_N \tag{30}$$

Now, rearranging the total residual as follows:

$$\begin{aligned} R_i &= \Big\{ \left(R_{i\,,ad}\right)_P - \left(R_{i\,,inv}\right)_N - \left(R_{i\,,vis}\right)_N \\ &+ \sum_J \left[\left(1 - B_j^i\right) \left(R_{j\,,inv}\right)_P T_j^i - \left(R_{j\,,ad}\right)_N T_j^i \right] \Big\} \tilde{\nu}_i \\ &+ \sum_J \left[B_j^i \left(R_{j\,,inv}\right)_P + \left(R_{j\,,vis}\right)_P \right] \tilde{\nu}_j \end{aligned} \tag{31}$$

where

$$T_j^i = \frac{\tilde{\nu}_j}{\tilde{\nu}_i} \tag{32}$$

and

$$B_j^i = \frac{T_j^i}{1 + T_j^i} = \frac{\tilde{\nu}_j}{\tilde{\nu}_j + \tilde{\nu}_i} \tag{33}$$

It should be emphasized that the formulation of the residual R_i as given in Eq. 31 is **algebraically identically** to the formulation given in Eq. 28 or Eq. 4.

For the sake of clarity we define $T^j_{i\ ad}$ and $T^j_{i\ inv}$ as follows:

$$T_j^i \equiv T^i_{j\ ad} = T^i_{j\ inv}\,. \tag{34}$$

Therefore Eq. 31 could be written as follows

$$\begin{aligned} R_i &= \Big\{ \left(R_{i\,,ad}\right)_P - \left(R_{i\,,inv}\right)_N - \left(R_{i\,,vis}\right)_N \\ &+ \sum_J \left[\underbrace{\left(1 - B_j^i\right) \left(R_{j\,,inv}\right)_P T^i_{j\ inv}}_{R_{od}} - \left(R_{j\,,ad}\right)_N T^i_{j\ ad} \right] \Big\} \tilde{\nu}_i \\ &+ \sum_J \left[B_j^i \left(R_{j\,,inv}\right)_P + \left(R_{j\,,vis}\right)_P \right] \tilde{\nu}_j \end{aligned} \tag{35}$$

The formulation of R_i as given in Eq. 35 is used to derive the appropriate implicit operator.

3.5 Presentation of source term

The source term in the Spalart-Allmaras turbulence model is of the following form:

$$S_i = P\left(\tilde{\nu}_i\right) - D\left(\tilde{\nu}_i\right) \tag{36}$$

$$= \hat{P}\left(\tilde{\nu}_i\right)\tilde{\nu}_i - \hat{D}\left(\tilde{\nu}_i\right)\tilde{\nu}_i. \tag{37}$$

In above equations, P and D denote the production and destruction terms respectively involved in Spalart–Allmaras turbulence model.

3.6 Implicit formulation

The backward Euler time stepping procedure, given in Eq. 2, for cell i can be written as follows:

$$\left[\frac{\Omega_i}{\Delta t} - \frac{\partial R_i}{\partial \tilde{\nu}_i} - \Omega_i \frac{\partial S_i}{\partial \tilde{\nu}_i}\right] \Delta_t \tilde{\nu}_i - \frac{\partial R_i}{\partial \tilde{\nu}_j} \Delta \tilde{\nu}_j = R_i + \Omega_i S_i \tag{38}$$

with the help of Eq. 35 the Jacobian, $\frac{\partial R_i}{\partial \tilde{\nu}_i}$, $\frac{\partial R_i}{\partial \tilde{\nu}_j}$ could be appropriately approximate to form the desired M-matrix.

In deriving the $\frac{\partial R_i}{\partial \tilde{\nu}_i}$ we use the following assumptions (refer to Eq. 35):

- $\left(R_{i\,,ad}\right)_P$, $\left(R_{i\,,inv}\right)_N$, $\left(R_{i\,,vis}\right)_N$ are frozen in time.
- $\left(1 - B^i_j\right)$, $T^i_{j\;ad}$, $\left(R_{i\,,ad}\right)_N$ are frozen in time.
- $T^i_{j\;inv}$ is differentiate with respect to $\tilde{\nu}_i$ **only**.
- $\tilde{\nu}_i$ that multiply R_{od} is frozen in time.
- Any positive contribution to $\frac{\partial R_i}{\partial \tilde{\nu}_i}$ is neglected.

Based on these assumptions the final form of $\frac{\partial R_i}{\partial \tilde{\nu}_i}$ is given as

$$\begin{aligned}\frac{\partial R_i}{\partial \tilde{\nu}_i} &= -\left(R_{i\,,inv}\right)_N - \left(R_{i\,,vis}\right)_N - \sum_J \left[\left(R_{j\,,ad}\right)_N T^i_j\right] \\ &- \sum_J \left[\left(1 - B^i_j\right)\left(R_{j\,,inv}\right)_P T^i_j\right]\end{aligned} \tag{39}$$

In deriving the $\frac{\partial R_i}{\partial \tilde{\nu}_j}$ we use the following assumptions:

- $\left(R_{j\,,inv}\right)_P$, $\left(R_{j\,,vis}\right)_P$, B^i_j are frozen in time.

Based on the above the final form of $\frac{\partial R_i}{\partial \tilde{\nu}_j}$ is given as

$$\frac{\partial R_i}{\partial \tilde{\nu}_j} = \sum_J \left[B^i_j \left(R_{j\,,inv}\right)_P + \left(R_{j\,,vis}\right)_P\right] \tag{40}$$

Finally the contribution of the source term to the LHS of Eq. 38 is given as follows:

$$-\frac{\partial S_i}{\partial \tilde{\nu}_i} = Max\left[\hat{D}_i - \hat{P}_i, 0\right] + Max\left[\left(\frac{\partial \hat{D}_i}{\partial \tilde{\nu}_i} - \frac{\partial \hat{P}_i}{\partial \tilde{\nu}_i}\right)\tilde{\nu}_i, 0\right]. \tag{41}$$

The unconditionally positive implicit procedure presented in this work is implemented in the flow solver High Resolution Flow Solver on Unstructured Meshes (HiFUN) [8] which is based on unstructured data based cell centre finite volume formulation.

4 RESULTS AND DISCUSSION

The efficacy of the implicit procedure presented in this work is demonstrated with the help of two high lift configurations. Computing flow past high lift configuration is still a challenge to present day CFD due to associated complexities in terms of geometry and flow physics. For solving RANS equations, Roe scheme [9] is used for inviscid flux computations, Green–Gauss theorem based diamond path reconstruction [10] is used for viscous flux computations, Venkatakrishnan limiter [11] is used for gradients limiting and matrix free symmetric Gauss Seidel (SGS) [12] procedure is used for implicit state update. In all the computations, the flow is assumed to be fully turbulent.

The first configuration is referred to as NHLP2D [13] configuration. This configuration has a slat, a main element and a flap. The free stream Mach number (M_∞) is 0.197 and free stream Reynolds number (Re_∞) is 3.52 millions. Figure 1 depicts the hybrid grid around the configuration consisting of triangular and quadrilateral elements. The total number of cells in the computational domain are 130,241. Figure 2 depicts the Mach fill plot at $\alpha = 21^o$. Tables 1 and 2 give the comparison of lift and drag coefficients at $\alpha = 12^o$ and $\alpha = 21^o$ respectively obtained using present computations with experimental results [13] and standard computations by Rumsey [14]. Figure 3 depicts the convergence

	Lift coefficient	Drag coefficient
Experiments	3.2023	0.0352
Present	3.1690	0.0408
Rumsey	3.2100	0.0386

Table 1: NHLP2D: C_L, C_D comparison, $\alpha = 12^o$

	Lift coefficient	Drag coefficient
Experiments	4.1335	0.0925
Present	4.1100	0.0764
Rumsey	4.2011	0.0720

Table 2: NHLP2D: C_L, C_D comparison, $\alpha = 21^o$

of density and $\tilde{\nu}$ residues at $\alpha = 12^o$. From this figure it can be seen that density residue converges to 10 decades and $\tilde{\nu}$ residue converges to about 8 decades. Figure 4 depicts the variation of CFL number with the iterations during convergence to steady state. This figure clearly brings out the robustness of proposed implicit procedure in terms of using large CFD number. Figures 5 and 6 depict the convergence of density/$\tilde{\nu}$ residue and variation of CFL number with iterations respectively at $\alpha = 21^o$. In this case also high convergence level for $\tilde{\nu}$ and usage of large CFL number are clearly evident.

The second configuration is referred to as OMAR 5–elements [15] configuration. This configuration consists of a slat, a main element, a primary flap and two auxiliary flaps. The free stream Mach number (M_∞) is 0.201 and free stream Reynolds number (Re_∞)

is 2.83 millions. Figure 7 depicts the unstructured quadrilateral grid around the configuration. The total number of cells in the computational domain are 162,578. Figure 8 depicts the Mach fill plot at $\alpha = 8^o$. Figures 9 and 10 depict the comparison of lift and drag coefficients at various angles of attack obtained using present computations with experimental results [15]. Figures 11 to 18 depict the convergence of density residue, $\tilde{\nu}$ residue and variation of CFL number with iterations for α varying from -4^o to 8^o. From these figures, it is clear that density residue converges to 10 decades and $\tilde{\nu}$ residue converges to about 8 decades. These results also demonstrate the efficacy of present implicit procedure in achieving high convergence level for $\tilde{\nu}$ and usage of large CFL number.

5 CONCLUSIONS

In the present work, we have proposed a robust implementation of Spalart–Allmaras turbulence model for unstructured grid using a positive implicit procedure. The implicit procedure is based on designing the associated implicit matrix such that it is M–matrix. The implicit procedure employs a unified treatment for the implicit operator of convection, diffusion, anti–diffusion and source terms involved in Spalart–Allmaras model equation. This implicit procedure guarantees positivity of modified turbulent viscosity ($\tilde{\nu}$) without the use of any clipping. The efficacy of the implicit procedure is demonstrated with the help of two high lift configurations. From the results presented in this paper, it is evident that the present implicit procedure is capable of not only achieving high level of convergence for turbulent viscosity but also using large CFL number to accelerate convergence to steady state.

REFERENCES

[1] P. R. Spalart and S. R. Allmaras, A one–equation turbulence model for aerodynamic flows, **AIAA Paper 92–0439** (1992).

[2] D. C. Wilcox, Reassessment of the scale–determining equation for advanced turbulence models, *AIAA Journal*, **26–11** (1988).

[3] F. R. Menter, Zonal two equation k–ω turbulence models for aerodynamic flows, 24^{th} *AIAA Fluid Dynamics Conference*, Orlando, FL, **AIAA paper 93–2906** (1993).

[4] J. C. Kok, Resolving the dependence on freestream values for the k–ω turbulence model, *AIAA Journal*, **38–7** (2000).

[5] Y. Mor–Yossef and Y. Levy, Unconditionally positive implicit procedure for two–equation turbulence models: Application to k–ω turbulence models, *Journal of Computational Physics*, **220**, 88–108 (2006).

[6] Y. Mor–Yossef and Y. Levy, Positive and Convergent Procedure for Non–linear kω Turbulence Models, 18^{th} *AIAA Computational Fluid Dynamics Conference*, **AIAA 2007–3962**, (2007).

[7] Y. Mor–Yossef and Y. Levy, Designing a positive second–order implicit time integration procedure for unsteady turbulent flows, *Comput. Methods Appl. Mech. Engg.*, **196**, 4196–4206 (2007).

[8] Nikhil Vijay Shende, Development of a general purpose flow solver for Euler equations, **Ph.D. Thesis**, Department of Aerospace Engineering, Indian Institute of Science (2005).

[9] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *Journal of Computational Physics*, **43**, 357 (1981).

[10] P. Jawahar and Hemant Kamath, A high–resolution procedure for Euler and Navier–Stokes computations on unstructured grids, *Journal of Computational Physics*, **164–1**, 165–203 (2000).

[11] Venkatakrishnan V., Convergence to Steady State Solutions of the Euler Equations on Unstructured Grids with Limiters, *Journal of Computational Physics*, **118**, 120–130 (1995).

[12] Shende N. and Balakrishnan N., A new migratory memory algorithm (MMA) for unstructured implicit finite volume codes, *AIAA Journal*, **42–9**, 1863–1870 (2004).

[13] Fejtek I., Summary of code validation results for a multielement aerofoil test case, AIAA Paper–**97–1932** (1997).

[14] Christopher Rumsey and Susan X. Ying, Prediction of high lift: review of present CFD capability, *Progress in Aerospace Sciences*, **38**, 145–180 (2002).

[15] E. Omar, T. Zierten, hi. Hahn, E. Szpiro, and A. hiabal, Two dimensional wind tunnel tests of a NASA supercritical aerofoil with various high lift systems, **NASA CR 2215, Volume 2–Test data** (1973).

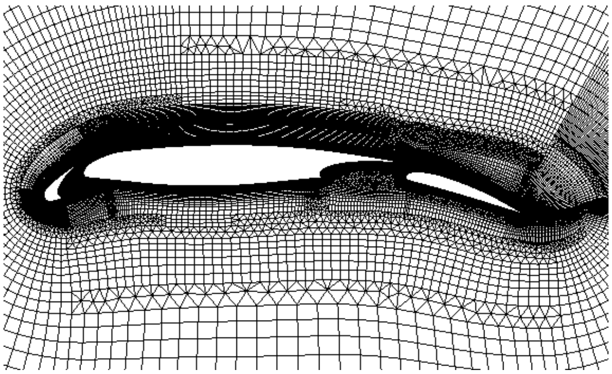

Figure 1: NHLP2D: Grid

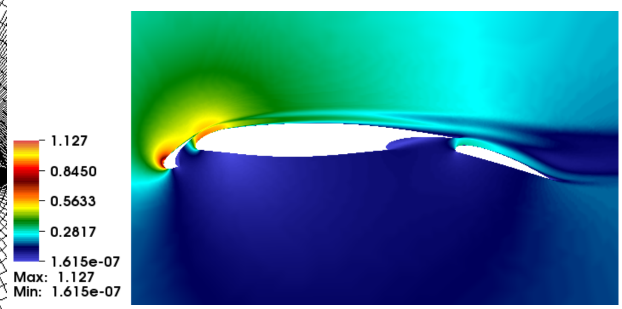

Figure 2: NHLP2D: Mach fill, $\alpha = 21^o$

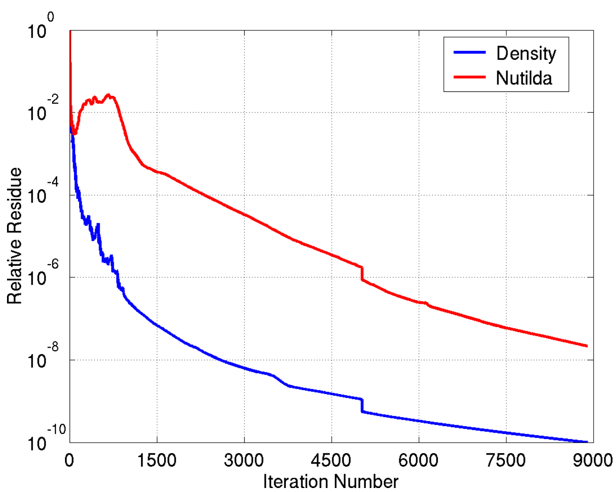

Figure 3: NHLP2D: Convergence, $\alpha = 12^o$

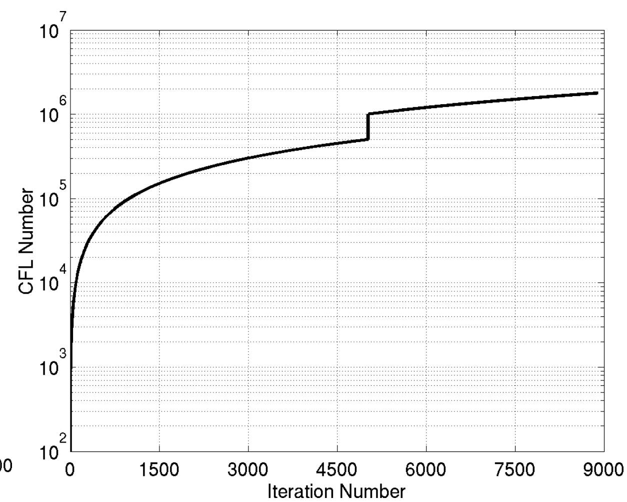

Figure 4: NHLP2D: CFL variation, $\alpha = 12^o$

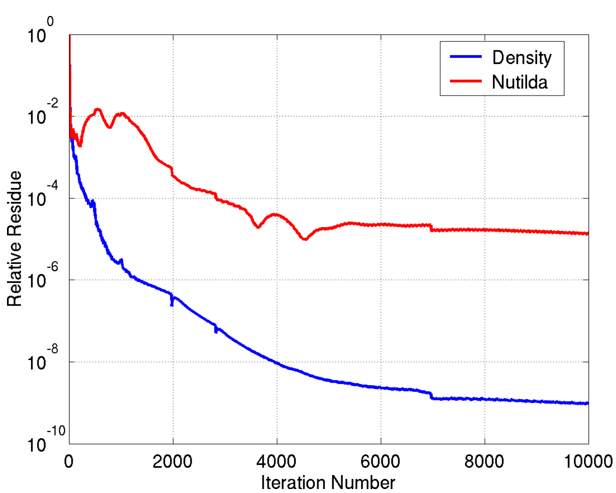

Figure 5: NHLP2D: Convergence, $\alpha = 21^o$

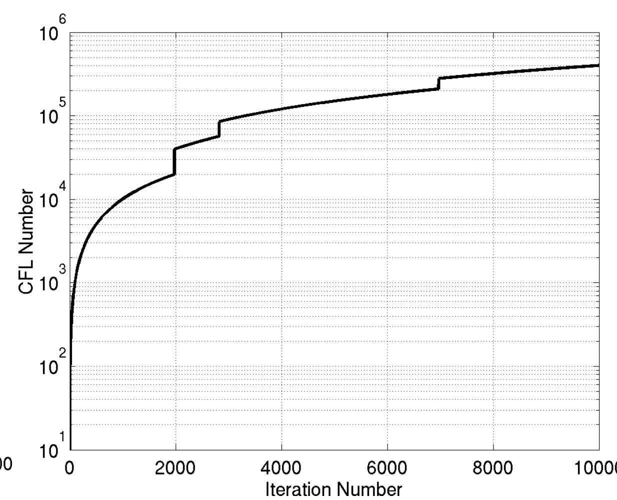

Figure 6: NHLP2D: CFL variation, $\alpha = 21^o$

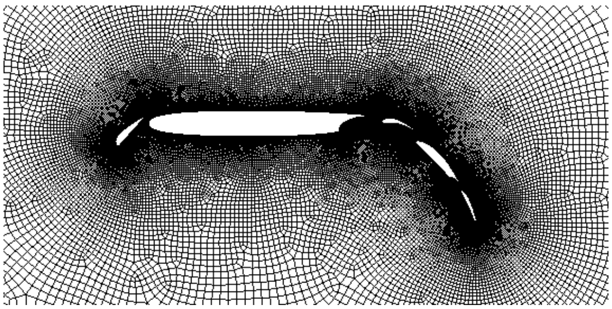

Figure 7: OMAR5: Grid

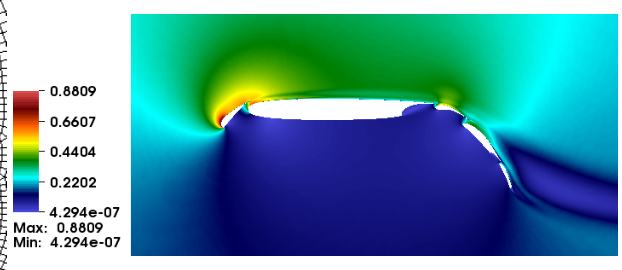

Figure 8: OMAR5: Mach fill, $\alpha = 8^o$

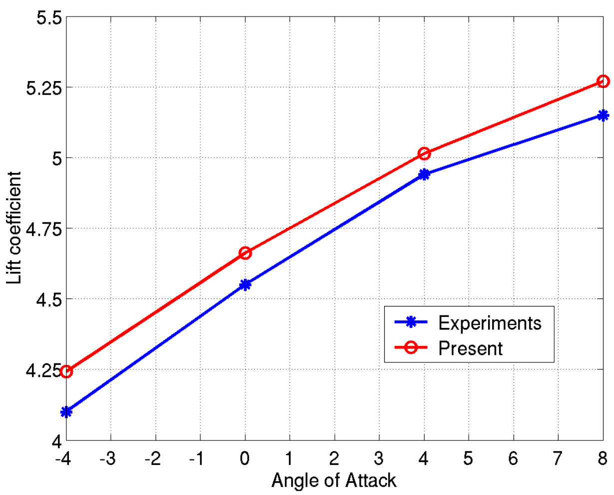

Figure 9: OMAR5: C_L comparison

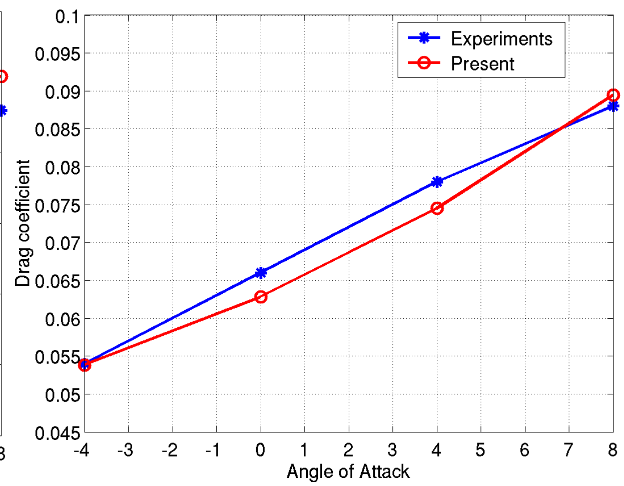

Figure 10: OMAR5: C_D comparison

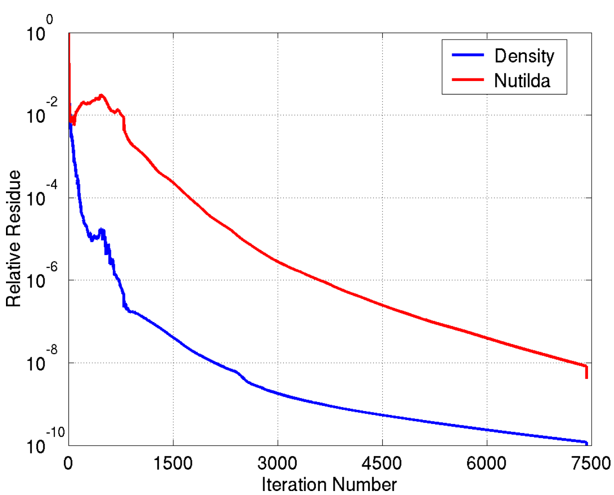

Figure 11: OMAR5: Convergence, $\alpha = -4^o$

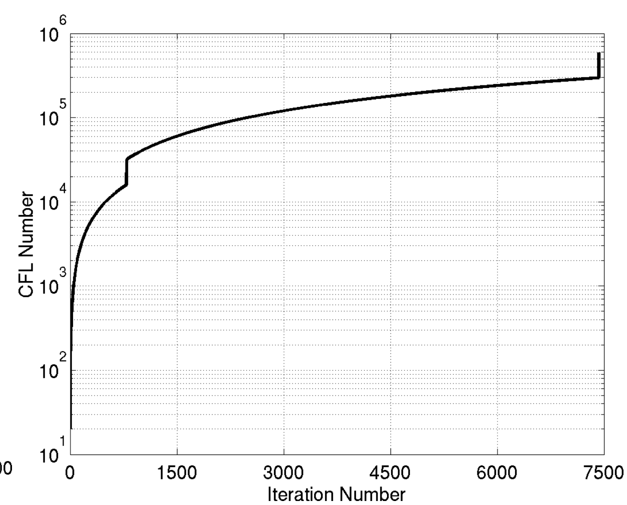

Figure 12: OMAR5: CFL variation, $\alpha = -4^o$

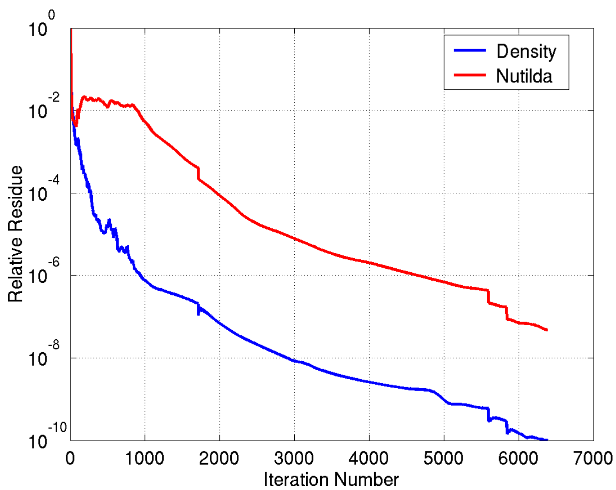

Figure 13: OMAR5: Convergence, $\alpha = 0^o$

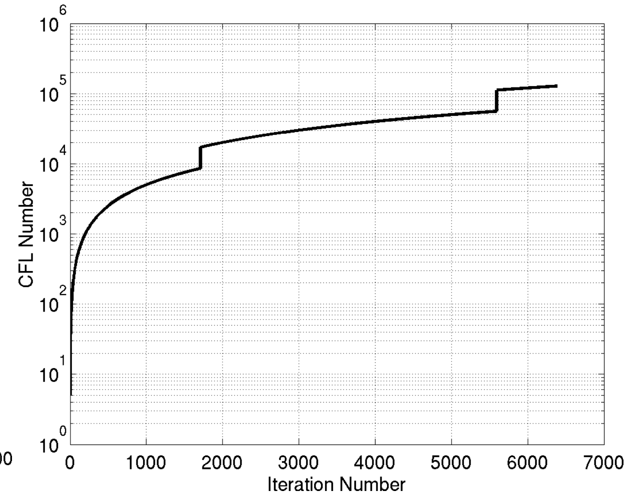

Figure 14: OMAR5: CFL variation, $\alpha = 0^o$

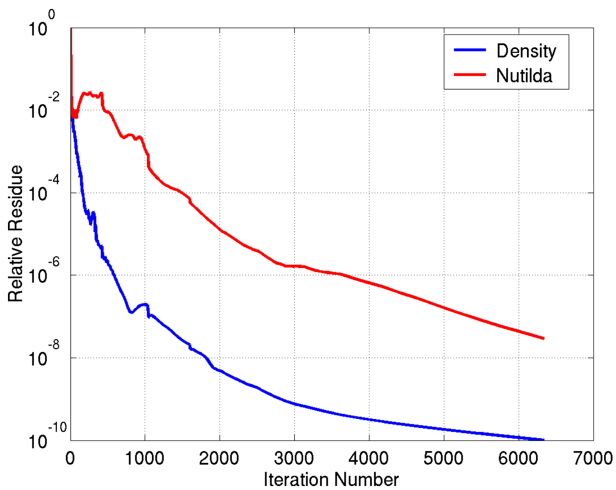

Figure 15: OMAR5: Convergence, $\alpha = 4^o$

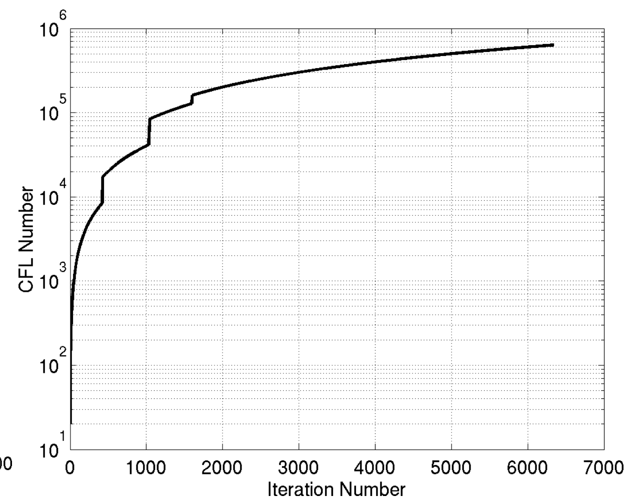

Figure 16: OMAR5: CFL variation, $\alpha = 4^o$

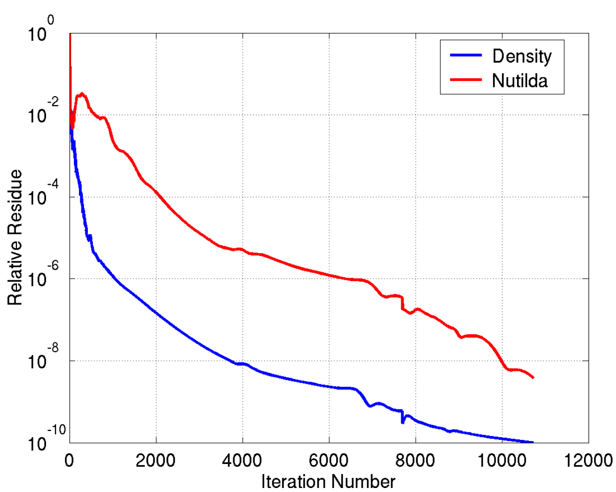

Figure 17: OMAR5: Convergence, $\alpha = 8^o$

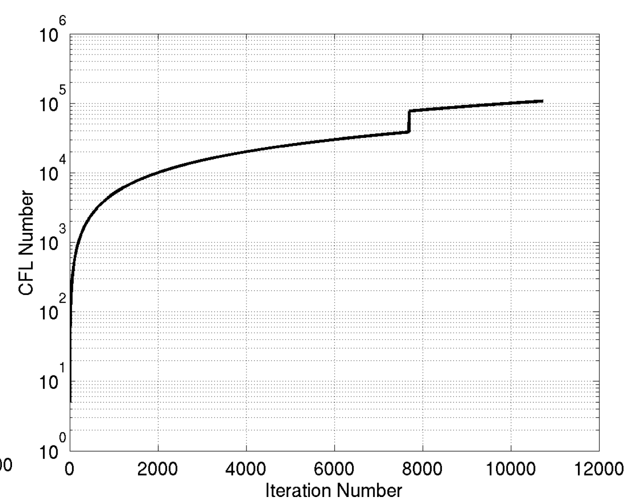

Figure 18: OMAR5: CFL variation, $\alpha = 8^o$